

6/12



Fig-9A

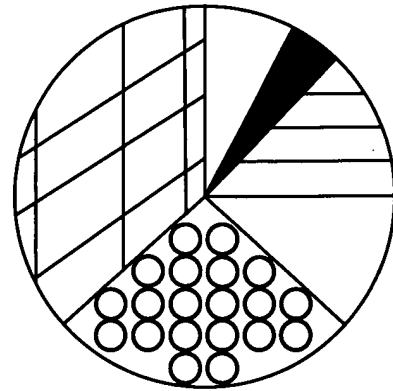


Fig-9B

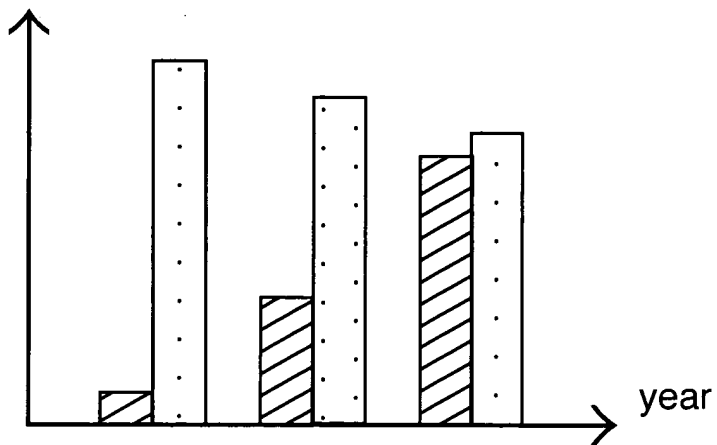


Fig-9C

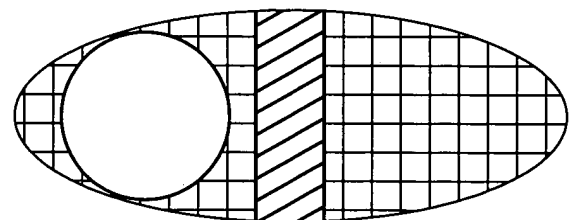


Fig-9D

8/12

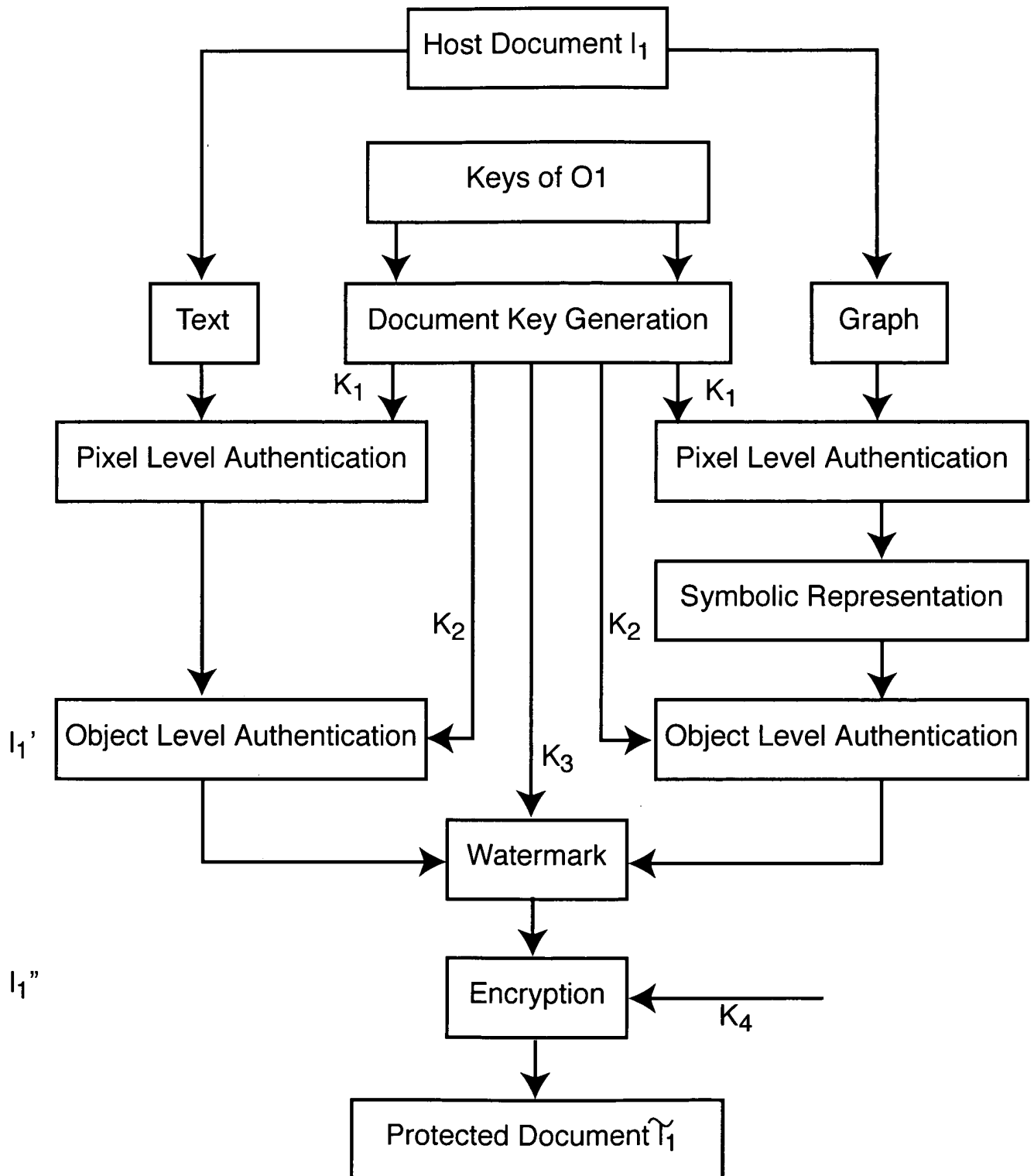


Fig-11

9/12

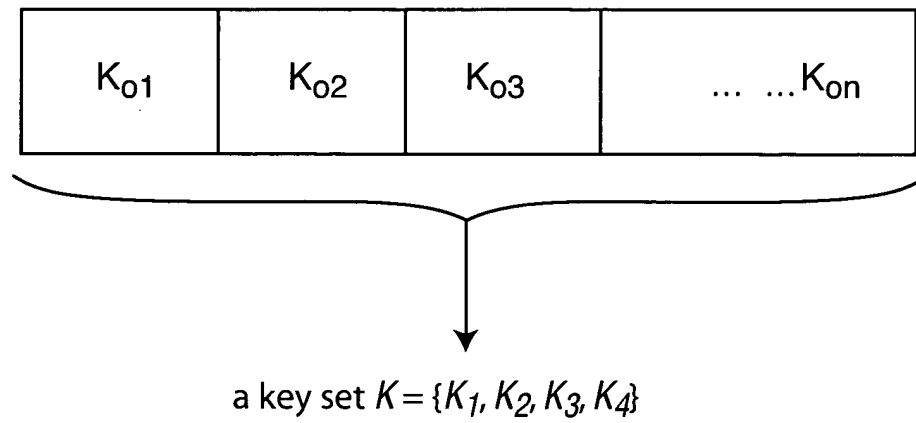


Fig-12

10/12

Relationship symbols	
<>	a tuple
∩	and
∪	or
≠	not
→	parent → child
⇨	sibling relation
↕	twin relation
←	child ← parent
>	contain relation
	condition
.	.
.	.
.	.
:	unconnected
Specification symbols	
&	size
#	shape
@	position
©	color

Fig-13

11/12

Fig-15

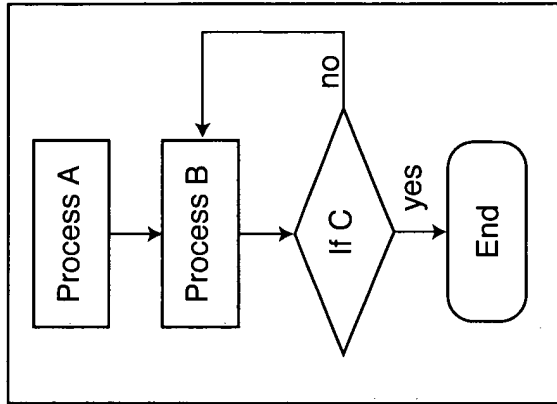


Fig-14

The system flow diagram is illustrated below. It shows the simplicity of the algorithm... "<N1{Process A', #1, &reg, @mid}→ N2{Process B', #1, &reg, @mid}→ N3{If C', #3, &reg, @mid}→< N4{End', #2, &reg, @mid}}yes; N2Ino>>"

(Text)	W/o content-dependent one way hash		Our algorithms, w/ content-dependent one way hash	
	Traditional line spacing	Traditional serif length	Coalescing	Object level
Special coding	Needed	Needed	May or may not needed	May or may not needed
Imperceptibility	Good	Good	OK	Good
Detectability	Bad	Bad	OK	Good
Pixel-level detectability	Bad	Bad	Good if Method I OK if Method II	Can't detect
Localization-ability	Bad	Some bad. Some OK	OK	Good
Copy and print	Bad	Bad	Good if Method I, bad if Method II	Good
Noise resistance-ability	Bad	OK	Good if Method I, bad if Method II	Good
Robustness to scaling	Good	OK	Good if Method I, bad if Method II	Good
				Duel level with coalescing
				May or may not needed
				Good
				Good
				OK
				Good
				Good
				Good
				Good

Fig-18

12/12

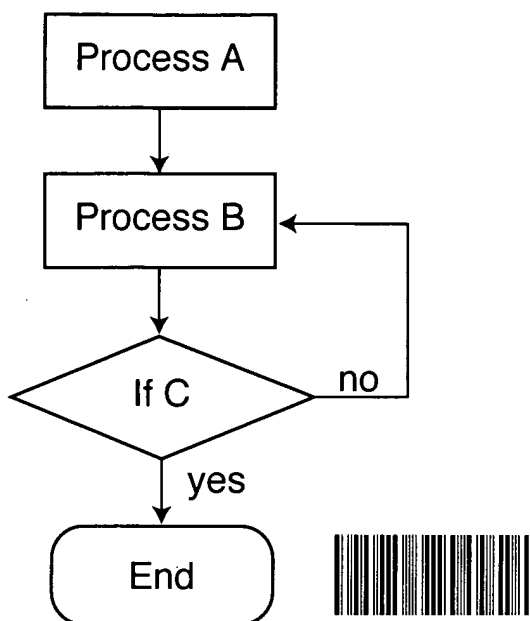


Fig-16

auth  
(a) Original size

auth  
(b) Enlarged

yes  
(c) Original size

yes  
(d) Enlarged

Fig-17